R Notebook

Decision Tree

```
library(pander)
```

```
df_master <- read.csv2("C:/Users/Marco/Desktop/df_master.csv", sep = ",")
```

```
str(df_master)
```

```
## 'data.frame':    1111503 obs. of  26 variables:
##  $ ID_EVENT_S          : int  12863255 12863259 12863262 12863264 12863273 12863277 12863279 1286329
##  $ TARGET              : int  1 1 0 0 1 1 1 1 1 1 ...
##  $ NUM_SEND_PREV       : int  1 2 1 1 1 1 5 2 4 3 ...
##  $ NUM_OPEN_PREV       : int  1 2 0 0 1 1 4 2 1 3 ...
##  $ NUM_CLICK_PREV      : int  1 0 0 0 0 0 1 0 0 2 ...
##  $ NUM_FAIL_PREV       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ OPEN_RATE_PREV      : Factor w/ 60 levels "0","0.08333333",..: 60 60 1 1 60 60 48 60 14 60 ...
##  $ CLICK_RATE_PREV     : Factor w/ 39 levels "0","0.08333333",..: 39 1 1 1 1 1 14 1 1 31 ...
##  $ W_SEND_PREV         : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
##  $ W_FAIL_PREV         : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ SEND_WEEKDAY        : Factor w/ 7 levels "domenica","giovedì",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ ID_NEG              : int  1 1 1 1 1 1 13 1 49 1 ...
##  $ TYP_CLI_FID         : int  1 1 1 1 1 1 0 1 1 1 ...
##  $ COD_FID             : Factor w/ 4 levels "PREMIUM","PREMIUM BIZ",..: 3 3 3 3 3 3 1 3 3 3 ...
##  $ STATUS_FID          : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ NUM_FIDs            : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ AGE_FID             : int  24 4 3 17 3 3 47 35 201 90 ...
##  $ W_PHONE             : int  1 1 1 1 1 1 NA 1 1 1 ...
##  $ TYP_CLI_ACCOUNT     : int  4 4 4 4 4 4 4 4 4 4 ...
##  $ TYP_JOB             : Factor w/ 14 levels "Altro","Artigiano",..: NA NA NA NA NA NA 9 NA NA NA ..
##  $ EMAIL_PROVIDER_CLEAN: Factor w/ 10 levels "alice.it","gmail.com",..: 2 2 2 2 10 6 4 2 2 2 ...
##  $ PRV                 : Factor w/ 110 levels "(Missing)","AG",..: 71 55 106 52 101 1 83 18 102 18 .
##  $ REGION              : Factor w/ 21 levels "(Missing)","ABRUZZO",..: 7 10 21 10 7 1 8 10 10 10 ...
##  $ FLAG_PRIVACY_1      : int  1 1 0 1 0 1 1 1 1 1 ...
##  $ FLAG_PRIVACY_2      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ FLAG_DIRECT_MKT     : int  1 1 0 1 1 1 1 0 1 1 ...
```

```
colonne_na <- sapply(colnames(df_master[,-c(1,6,7,8,9,10,21,22,23)]),
                     function(x) any(is.na(df_master[,x])))
#colonne_na[colonne_na == TRUE]
total_rf <- df_master[,-c(1,6,7,8,9,10,21,22,23)][,!colonne_na]
```

```
str(total_rf)
```

```
## 'data.frame':    1111503 obs. of  15 variables:
##  $ TARGET         : int  1 1 0 0 1 1 1 1 1 1 ...
##  $ NUM_SEND_PREV  : int  1 2 1 1 1 1 5 2 4 3 ...
##  $ NUM_OPEN_PREV  : int  1 2 0 0 1 1 4 2 1 3 ...
##  $ NUM_CLICK_PREV : int  1 0 0 0 0 0 1 0 0 2 ...
##  $ SEND_WEEKDAY   : Factor w/ 7 levels "domenica","giovedì",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ ID_NEG         : int  1 1 1 1 1 1 13 1 49 1 ...
##  $ TYP_CLI_FID    : int  1 1 1 1 1 1 0 1 1 1 ...
##  $ COD_FID        : Factor w/ 4 levels "PREMIUM","PREMIUM BIZ",..: 3 3 3 3 3 3 1 3 3 3 ...
##  $ STATUS_FID     : int  1 1 1 1 1 1 1 1 1 1 ...
```

```
##  $ NUM_FIDs       : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ AGE_FID        : int  24 4 3 17 3 3 47 35 201 90 ...
##  $ TYP_CLI_ACCOUNT: int  4 4 4 4 4 4 4 4 4 4 ...
##  $ FLAG_PRIVACY_1 : int  1 1 0 1 0 1 1 1 1 1 ...
##  $ FLAG_PRIVACY_2 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ FLAG_DIRECT_MKT: int  1 1 0 1 1 1 1 0 1 1 ...
```

```
total_rf['TARGET'] <- as.factor(df_master['TARGET'])
total_rf['NUM_SEND_PREV'] <- as.factor(df_master['NUM_SEND_PREV'])
total_rf['NUM_OPEN_PREV'] <- as.factor(df_master['NUM_OPEN_PREV'])
total_rf['NUM_CLICK_PREV'] <- as.factor(df_master['NUM_CLICK_PREV'])
total_rf['NUM_FAIL_PREV'] <- as.factor(df_master['NUM_FAIL_PREV'])
total_rf['ID_NEG'] <- as.factor(df_master['ID_NEG'])
total_rf['TYP_CLI_FID'] <- as.factor(df_master['TYP_CLI_FID'])
total_rf['STATUS_FID'] <- as.factor(df_master['STATUS_FID'])
total_rf['NUM_FIDs'] <- as.factor(df_master['NUM_FIDs'])
#total_rf['AGE_FID'] <- as.numeric(df_master['AGE_FID'])
total_rf['TYP_CLI_ACCOUNT'] <- as.factor(df_master['TYP_CLI_ACCOUNT'])
total_rf['FLAG_PRIVACY_1'] <- as.factor(df_master['FLAG_PRIVACY_1'])
total_rf['FLAG_PRIVACY_2'] <- as.factor(df_master['FLAG_PRIVACY_2'])
total_rf['FLAG_DIRECT_MKT'] <- as.factor(df_master['FLAG_DIRECT_MKT'])
```

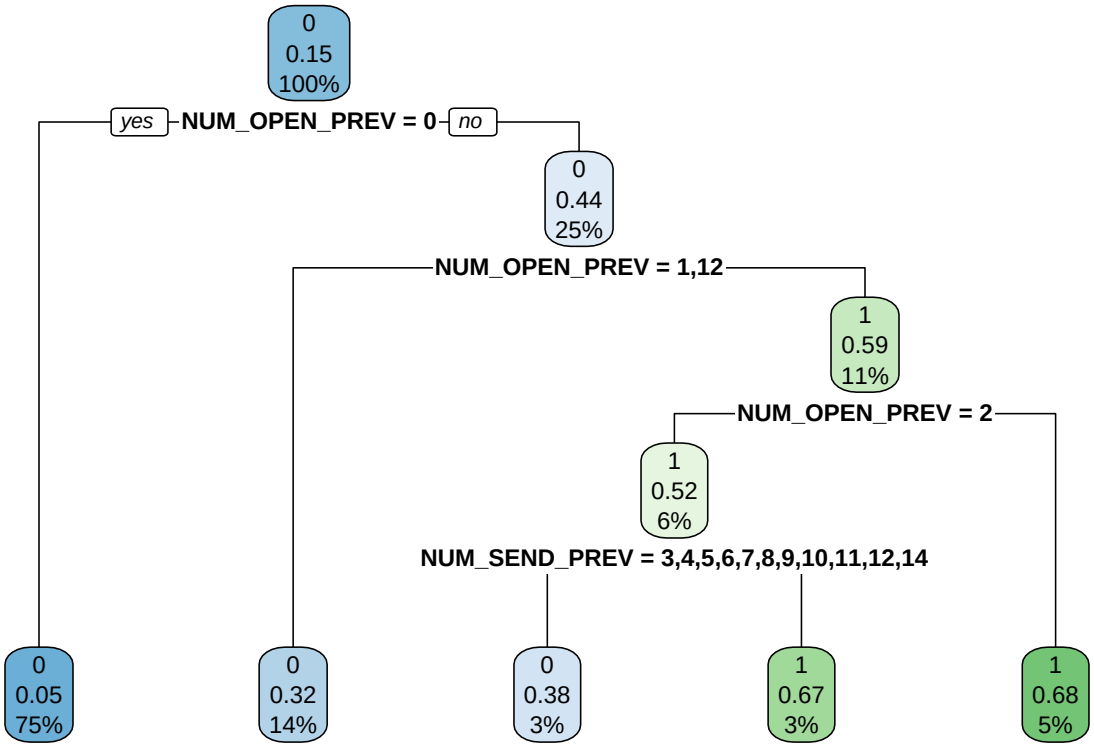
```
str(total_rf)
```

```
## 'data.frame':    1111503 obs. of  16 variables:
##  $ TARGET         : Factor w/ 2 levels "0","1": 2 2 1 1 2 2 2 2 2 2 ...
##  $ NUM_SEND_PREV  : Factor w/ 20 levels "0","1","2","3",..: 2 3 2 2 2 2 6 3 5 4 ...
##  $ NUM_OPEN_PREV  : Factor w/ 15 levels "0","1","2","3",..: 2 3 1 1 2 2 5 3 2 4 ...
##  $ NUM_CLICK_PREV : Factor w/ 7 levels "0","1","2","3",..: 2 1 1 1 1 1 2 1 1 3 ...
##  $ SEND_WEEKDAY   : Factor w/ 7 levels "domenica","giovedì",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ ID_NEG         : Factor w/ 49 levels "1","2","3","4",..: 1 1 1 1 1 1 13 1 49 1 ...
##  $ TYP_CLI_FID    : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 1 2 2 2 ...
##  $ COD_FID        : Factor w/ 4 levels "PREMIUM","PREMIUM BIZ",..: 3 3 3 3 3 3 1 3 3 3 ...
##  $ STATUS_FID     : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ NUM_FIDs       : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 1 1 ...
##  $ AGE_FID        : int  24 4 3 17 3 3 47 35 201 90 ...
##  $ TYP_CLI_ACCOUNT: Factor w/ 2 levels "2","4": 2 2 2 2 2 2 2 2 2 2 ...
##  $ FLAG_PRIVACY_1 : Factor w/ 2 levels "0","1": 2 2 1 2 1 2 2 2 2 2 ...
##  $ FLAG_PRIVACY_2 : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ FLAG_DIRECT_MKT: Factor w/ 2 levels "0","1": 2 2 1 2 2 2 2 1 2 2 ...
##  $ NUM_FAIL_PREV  : Factor w/ 11 levels "0","1","2","3",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```
set.seed(12345)
sample <- sample.int(n = nrow(total_rf), size = floor(.75*nrow(total_rf)), replace = F)
train <- total_rf[sample, ]
test  <- total_rf[-sample, ]
```

```
library(rpart)
library(rpart.plot)
```

```
tree_model <- rpart(TARGET ~ ., data= train)
rpart.plot(tree_model, extra = 106)
```




```
predict_unseen <-predict(tree_model, test, type = 'class')
```

```
table_mat <- table(test$TARGET, predict_unseen)
table_mat
```

```
##    predict_unseen
##          0      1
##   0 228839   6961
##   1  27085  14991
```

```
accuracy_Test <- sum(diag(table_mat)) / sum(table_mat)
accuracy_Test
```

```
## [1] 0.8774777
```

```
precision <- table_mat[2,2] / (table_mat[2, 2] +table_mat[1, 2])

precision
```

```
## [1] 0.6828991
```

```
recall <- table_mat[2,2] / (table_mat[2, 2] + table_mat[2, 1])

recall
```

```
## [1] 0.3562839
```

```
F_measure = (2*precision*recall) / (precision + recall)
```

```
F_measure
```

```
## [1] 0.4682639
```